

Практическое задание №11 (классы)

Необходимо реализовать класс `Matrix` – целочисленную прямоугольную матрицу со следующими методами:

1. Конструктором, получающим в качестве параметров количество строк и столбцов в матрице. В результате должна быть создана нулевая матрица.
2. Конструктором копирования.
3. Конструктором перемещения.
4. Деструктором.
5. Методом заполнения матрицы данными из файла.
6. Перегруженным оператором присваивания с копированием `operator =`.
7. Перегруженным оператором присваивания с перемещением `operator =`.
8. Перегруженным оператором сравнения двух матриц `operator ==` (реализовать в виде метода класса).
9. Перегруженным оператором суммирования двух матриц `operator +` (реализовать в виде дружественной функции). Исходные операнды изменяться не должны.
10. Перегруженным оператором индексирования `operator []`, который возвращает указатель на заданную строку матрицы.
11. Перегруженным оператором индексирования `operator ()` с двумя параметрами (индексами элемента матрицы), который контролирует корректность значений индексов и генерирует исключение `std::runtime_error` в случае выхода любого индекса за границы матрицы. Оператор должен возвращать `l-value`.
12. Методом, возвращающим транспонированную матрицу. Исходная матрица изменяться не должна.

При желании можно добавить другие методы.

Для хранения элементов матрицы следует использовать **динамически** созданный **двумерный** массив элементов типа `int` (`private`-поля в классе – указатель на двумерный динамический массив и две размерности).

Для сдачи проекта использовать структуру в файловой системе:

`gxxxxx/11/Task11/Matrix.h` – файл с объявлением класса.

`gxxxxx/11/Task11/Matrix.cpp` – файл с определением методов класса.

`gxxxxx/11/Task11.Tests/Matrix.Tests.cpp` – файлы с тестами пунктов задания.

Необходимые условия сдачи:

- стиль программирования должен соответствовать установленным правилам;
- программа должна быть протестирована с помощью `gtest`.